Problem L
Trio
Time Limit: 2 Seconds

Let A be any set of n natural numbers whose decimal representations consist of exactly four digits without 0 in any decimal place.

A *trio* is a set of three numbers $\{a, b, c\}$ chosen from A such that the following conditions are fulfilled simultaneously:

- The ones decimals of three numbers a, b, c are either all equal or all distinct.
- The tens decimals of three numbers a, b, c are either all equal or all distinct.
- The hundreds decimals of three numbers a, b, c are either all equal or all distinct.
- The thousands decimals of three numbers a, b, c are either all equal or all distinct.

For examples, $\{1425, 1113, 1354\}$ is a trio if the three numbers are members of A because the ones decimals of the three numbers are all distinct, their tens decimals are all distinct, their hundreds decimals are all distinct, and their thousands decimals are all equal. The set $\{1425, 1113, 5436\}$, however, is not a trio, even if A contains those three numbers.

Given a set A as input, write a program that computes and prints out the number of different trios.

Input
Your program is to read from standard input. The input starts with a line consisting of a single integer n ($1 \leq n \leq 2{,}000$) that represents the number of members in A. Each of the following n lines consists of a positive integer in decimal form that consists of exactly four digits without 0 in any decimal place. These n numbers are supposed to be all distinct and the members of the input set A.

Output
Your program is to write to standard output. Print exactly one line. The line should consists of a single integer that represents the number of different trios for the input set A.

The following shows sample input and output for two test cases.

Sample Input 1	Output for the Sample Input 1
6 1234 1235 1244 1233 7133 8133	1

Sample Input 2	Output for the Sample Input 2
9 1234 5678 9123 4567 8912 3456 7891 2345 6789	84